

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1-20. (Cancelled).

21. (Currently amended) The method of claim 45, ~~claim 20~~, wherein the step of verifying for two given objects is carried out only when a command is sent after the new common key was provided.

22.. (Currently amended) The method of claim 45, ~~claim 20~~, wherein the step of providing of the new common key comprises:
generating a new common key; and
transmitting the generated new common key.

23. (Previously presented) The method of claim 22, wherein the step of generation is carried out using a single object.

24. (Previously presented) The method of claim 22, wherein the step of generation is carried out using two objects.

25. (Previously presented) The method of claim 22, wherein the step of transmission comprises a point-to-multipoint transmission.

26. (Previously presented) The method of claim 22, wherein the step of transmission comprises point-to-point transmission.

27. (Previously presented) The method of claim 26, wherein the point-to-point transmission comprises an action by the user on each point.

28. (Previously presented) The method of claim 22, wherein the step of transmission comprises:

- a point-to-point transmission in a sub-group of the objects; and
- a point-to-multipoint transmission to another sub-group of the objects.

29. (Previously presented) The method of claim 22, wherein the transmission step comprises, when the new common key of an object is transmitted to another object, verification that the two objects contain the old common key.

30. (Currently amended) An operating program for a bidirectional object, contained in a memory, and adapted to store at least one common key and at least one piece of information on pairing, comprising:

- (a) a routine of receiving a ~~new~~ common key;
- (b) a routine ~~for~~ of learning and keeping in the memory the pairing information of identifiers of other objects to which the bidirectional object is paired;
- (c) a routine of receiving and storing a new common key without erasing the pairing information;
- (d) a routine of receiving a command from a paired transmitter object;
- ~~(d)(e)~~ (e) a routine of verifying ~~for a command received from a paired transmitter object~~, the presence of the new common key in the paired transmitter object upon receipt of the command from the paired transmitter object; ~~and~~
- ~~(e)(f)~~ (f) a routine of refusing to execute the command if the verification is negative, although the command is received from a paired object; and
- (g) a routine of executing the command if the verification is positive.

31. (Previously presented) The program of claim 30, wherein the routine of verifying for a given pairing is implemented only when a command is received.

32. (Previously presented) The program of claim 30, further comprising a routine of generating a new common key.

33. (Previously presented) The program of claim 32, wherein the routine of generating comprises a sub-routine of transmitting a command to generate the common key to another object.

34. (Previously presented) The program of claim 30, further comprising a routine of transmitting a new common key to another object.

35. (Previously presented) The program of claim 30, further comprising a routine of transmitting a new common key to more than one object.

36. (Currently amended) An operating program for a bidirectional object, contained in a memory, and adapted to store at least one common key and at least one piece of information on pairing, comprising:

- (a) a routine of receiving of a new-common key;
- (b) a routine for learning and keeping in the memory the pairing information of identifiers of other objects to which the bidirectional object is paired;
- (c) a routine of receiving and storing a new common key without erasing pairing information;
- ~~(e)~~(d) a routine of transmitting of a command to a targeted paired object; and
- ~~(d)~~(e) a routine of verifying the presence of the new common key in the targeted object.

37. (Previously presented) The program of claim 36, wherein the routine of verifying for a given pairing is implemented only when a command is transmitted.

38. (Previously presented) The program of claim 36, further comprising a routine of generating a new common key.

39. (Previously presented) The program of claim 38, wherein the routine of generating comprises a sub-routine of transmitting of a command to generate the common key to another object.

40. (Previously presented) The program of claim 36, further comprising a routine of transmitting of a new common key to another object.

41. (Previously presented) The program of claim 36, further comprising a routine of transmitting of a new common key to several other objects.

42. (Currently amended) A bidirectional object, having:
a receiving stage;
a transmitting stage;
a memory, containing an operating program for a bidirectional object adapted to store at least one common key and at least one piece of information on pairing, and

a control unit executing said program; said program comprising:
a routine adapted to receive a new common key;
a routine adapted to learn and keep in the memory the pairing information of identifiers of other objects to which the bidirectional object is paired;

a routine adapted to receive and store a new common key without erasing pairing information;

a routine adapted to receive a command from a paired transmitter object;

a routine adapted to verify ~~a command received from a paired transmitter object~~ of the presence of the new common key in the transmitter object upon receipt of the command from the paired transmitter object; and

a routine adapted to refuse execution of the command if the verification is negative, although the command is received from a paired object; and

a routine adapted to execute the command if the verification is positive.

43. (Currently amended) The method of claim 45, claim 20, wherein the new common key is provided globally to all the objects of the installation, without consideration of the pairings.

44. (Currently amended) The method of claim 45, claim 20, wherein pairings of paired objects are suspended during the step of providing the new common key and become valid again upon confirmation that the paired objects contain the new common key.

45. (New) A method for reprogramming a plurality of bidirectional objects belonging to an installation, each object of the installation having a univalent identifier stored in a memory, wherein the method comprises:

- i. providing an initial common key;
- ii. storing the initial common key in the memory of at least a first object, a second object, and a third object of a group of objects in the installation;
- iii. pairing the first object to the second object and to the third object, wherein the pairing of two objects includes storing in the memory of the objects to be paired information on the identifier of the other object;
- iv. providing a new common key and storing the new common key in the memory of the first object and in the memory of the second object, excluding the third object from being provided with the new common key; then

- v. keeping in the memory of the second and the third object the information on the identifier of the first object;
- vi. keeping in the memory of the first object the information on the identifier of the second object and the identifier of the third object;
- vii. sending a command from the third object to the first object;
- viii. receiving a command from the third object at the first object;
- ix. verifying, in the first object, if the new common key is stored in the memory of the third object;
- x. verifying, in the first object, if the information on the identifier of the third object is stored in the memory of the first object; and
- xi. refusing by the first object the execution of the command sent by the third object, although the first object and the third object are still paired, thereby providing for the third object to be excluded from the group without reprogramming the pairings existing between the objects of the installation.